

# Introduction to R Markdown

*Combine text, code, and output in one document*

by Martin Frigaard

Written: October 03 2021

Updated: November 30 2021

[Created using the "λέξις" theme](#)



Artwork by @allison\_horst

# Materials



Link to slides:

<https://mjfrigaard.github.io/csuc-data-journalism/slides.html>

Link to exercises:

<https://mjfrigaard.github.io/csuc-data-journalism/lessons-exercises.html>



# *What is RMarkdown?*

Three technologies:

- 1) Markdown is a plain text markup language for capturing **human-readable prose**
- 2) Data manipulation/graphing/statistical language engines for computing **machine-readable code**
- 3) Multiple **output options** for creating PDFs, Word docs, PowerPoints, HTML, etc.

# How R Markdown works

rmarkdown works directly with knitr

rmarkdown combines YAML, markdown, and R code into a markdown document and passes it to knitr



knitr uses pandoc (a universal document conversion tool) to generate the specified document format





# Exercises

We will create an example HTML report using the R Markdown template provided by RStudio

# Exercise 1: create a new RMarkdown file

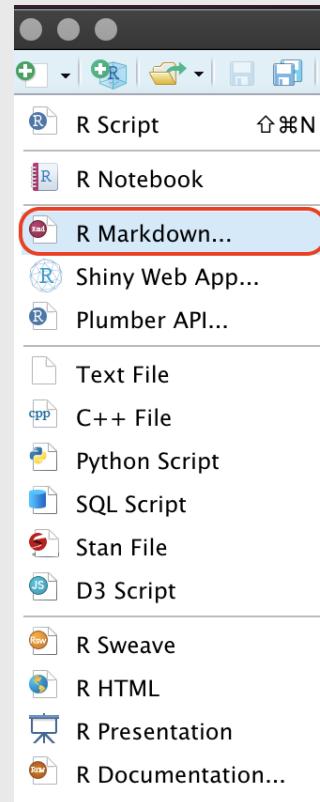


Click on **File** >

then **New File** >

then **R Markdown**

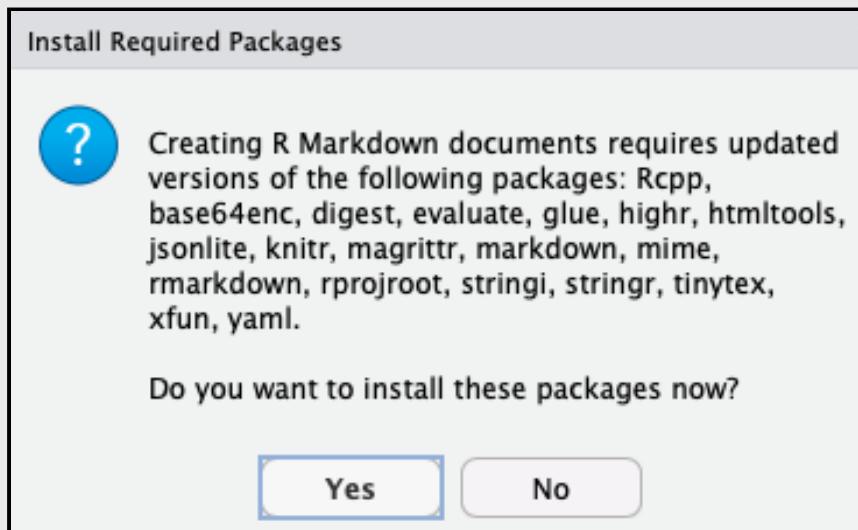
***Or use the drop-down menu***





# Install required packages

If you're in a fresh RStudio.Cloud session, you *might* be asked to install the required packages for R Markdown, Click **Yes**



You will see RStudio installing the packages in the **Jobs** pane

```
Console Terminal × Jobs ×
R Markdown Dependencies 0:07
* DONE (glue)

The downloaded source packages are in
  '/tmp/RtmpzyUUTc/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/highr_0.8.tar.gz'
Content type 'application/x-tar' length 40478 bytes (39 KB)
=====
downloaded 39 KB

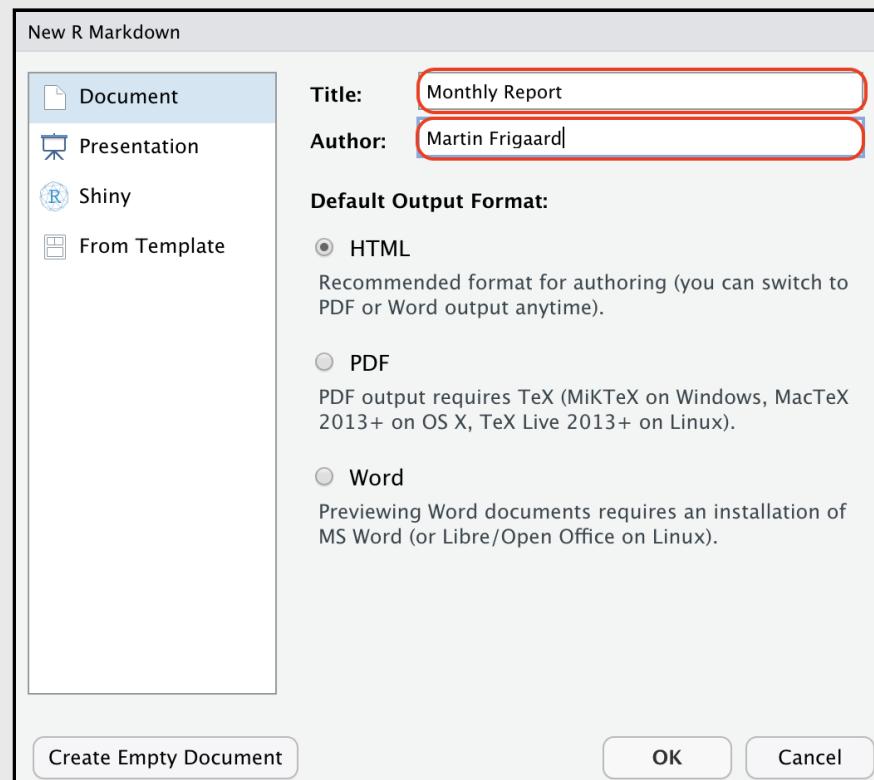
[7/21] Installing rlang...
* installing *binary* package 'highr' ...
* DONE (highr)

The downloaded source packages are in
  '/tmp/RtmpzyUUTc/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/rlang_0.4.8.tar.gz'
```



# New R Markdown File

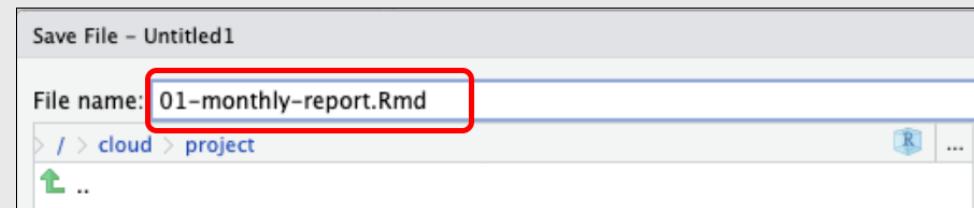
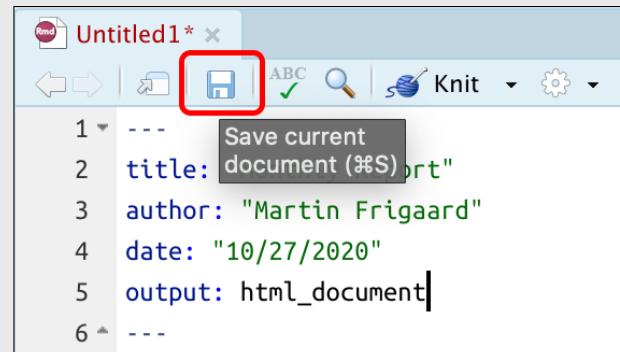
Enter '**Title**' and '**Author**' of your report and click **OK**





# Save your .Rmd file

Click on the small floppy disk, enter a name (with .Rmd extension), and save your .Rmd file

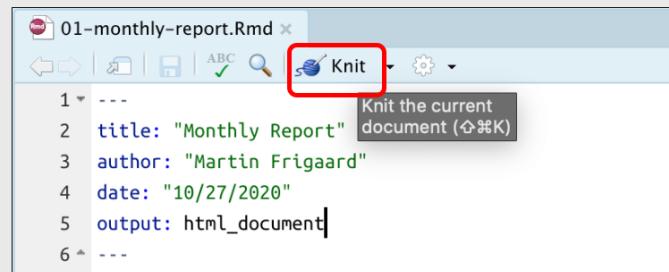
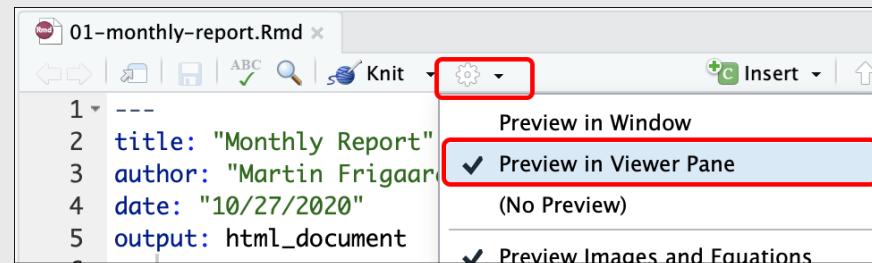




# Knit your .Rmd file

Click on the small gear, select **Preview in Viewer Pane**

Click on the knit icon (ball of yarn)



# Our First R Markdown Report!



Your Workspace / bmrn-04-rmd-intro

File Edit Code View Plots Session Build Debug Profile Tools Help

01-monthly-report.Rmd x

```
1- ---
2- title: "Monthly Report"
3- author: "Martin Frigaard"
4- date: "10/27/2020"
5- output: html_document
6- ---
7-
8- ````{r setup, include=FALSE}
9- knitr::opts_chunk$set(echo = TRUE)
10- ````

11- ## R Markdown
12-
13- This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.
14-
15- When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
16-
17- ````{r cars}
18- summary(cars)
19- ````

21- ## Including Plots
22-
23- You can also embed plots, for example:
24-
25- ````{r pressure, echo=FALSE}
26- plot(pressure)
27- ````

29-
30- Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.
31-
```

4:13 Monthly Report : R Markdown z

Environment History Connections Tutorial

Files Plots Packages Help Viewer

R 4.0.3 Martin J Frigaard

## Monthly Report

Martin Frigaard  
10/27/2020

### R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##   speed      dist
## Min. :4.0  Min. : 2.00
## 1st Qu.:12.0 1st Qu.:26.00
## Median :15.0 Median :36.00
## Mean   :15.4 Mean  :42.98
## 3rd Qu.:19.0 3rd Qu.:56.00
## Max.  :25.0  Max. :120.00
```

### Including Plots

You can also embed plots, for example:

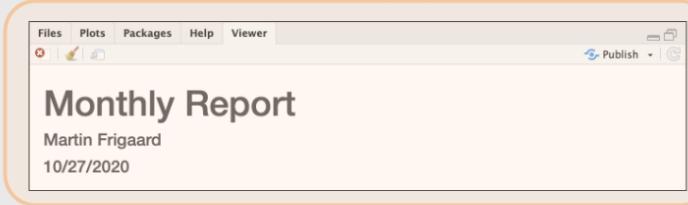
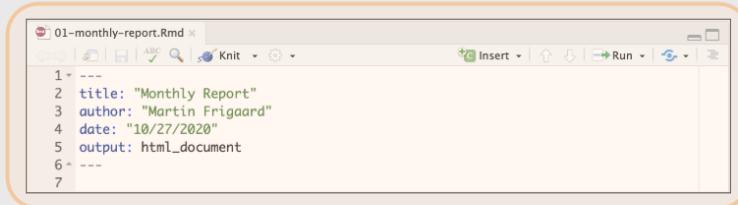
Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



# How R Markdown Works (under the hood)

# R Markdown is made up of three elements

YAML header = metadata



The left screenshot shows the RStudio interface with the file '01-monthly-report.Rmd' open. The code editor displays the YAML header:

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output: html_document  
6 ---  
7
```

The right screenshot shows the 'Viewer' pane displaying the generated HTML document titled 'Monthly Report' by Martin Frigaard on 10/27/2020.

Markdown = prose

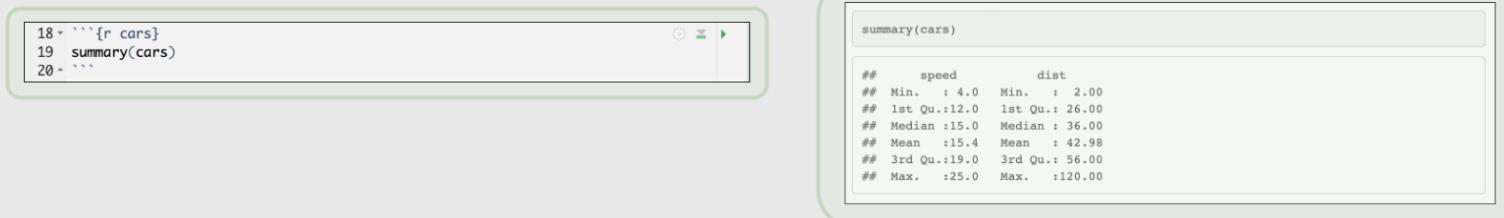


The left screenshot shows the RStudio code editor with R Markdown prose:

```
12 ## R Markdown  
13  
14 This is an R Markdown document. Markdown is a simple formatting syntax for  
authoring HTML, PDF, and MS Word documents. For more details on using R Markdown  
see <http://rmarkdown.rstudio.com>.  
15  
16 When you click the **Knit** button a document will be generated that includes both  
content as well as the output of any embedded R code chunks within the document.  
You can embed an R code chunk like this:
```

The right screenshot shows the RStudio viewer pane displaying the generated HTML document with the same R Markdown prose.

Code chunks = R code



The left screenshot shows the RStudio code editor with an R code chunk:

```
18 `r cars`  
19 summary(cars)  
20 `r`
```

The right screenshot shows the RStudio viewer pane displaying the output of the R code chunk, which is a summary statistics table for the 'cars' dataset:

	speed	dist
## Min.	4.0	2.00
## 1st Qu.	12.0	1st Qu.: 26.00
## Median	15.0	Median : 36.00
## Mean	15.4	Mean : 42.98
## 3rd Qu.	19.0	3rd Qu.: 56.00
## Max.	25.0	Max. :120.00

# Rmarkdown combines metadata, markdown, and R code



- `.yaml` = Metadata
- `.md` = Prose
- `.R` = Code

The result is a file framework for creating reproducible reports using YAML, Markdown, and computer code

# R Markdown: YAML



~~.yaml~~ = Metadata

~~.md~~ = Prose

~~.R~~ = Code

YAML is a human friendly data serialization standard for all programming languages.

YAML stands for '*YAML Ain't Markup Language*' (funny, huh?)

# R Markdown: YAML

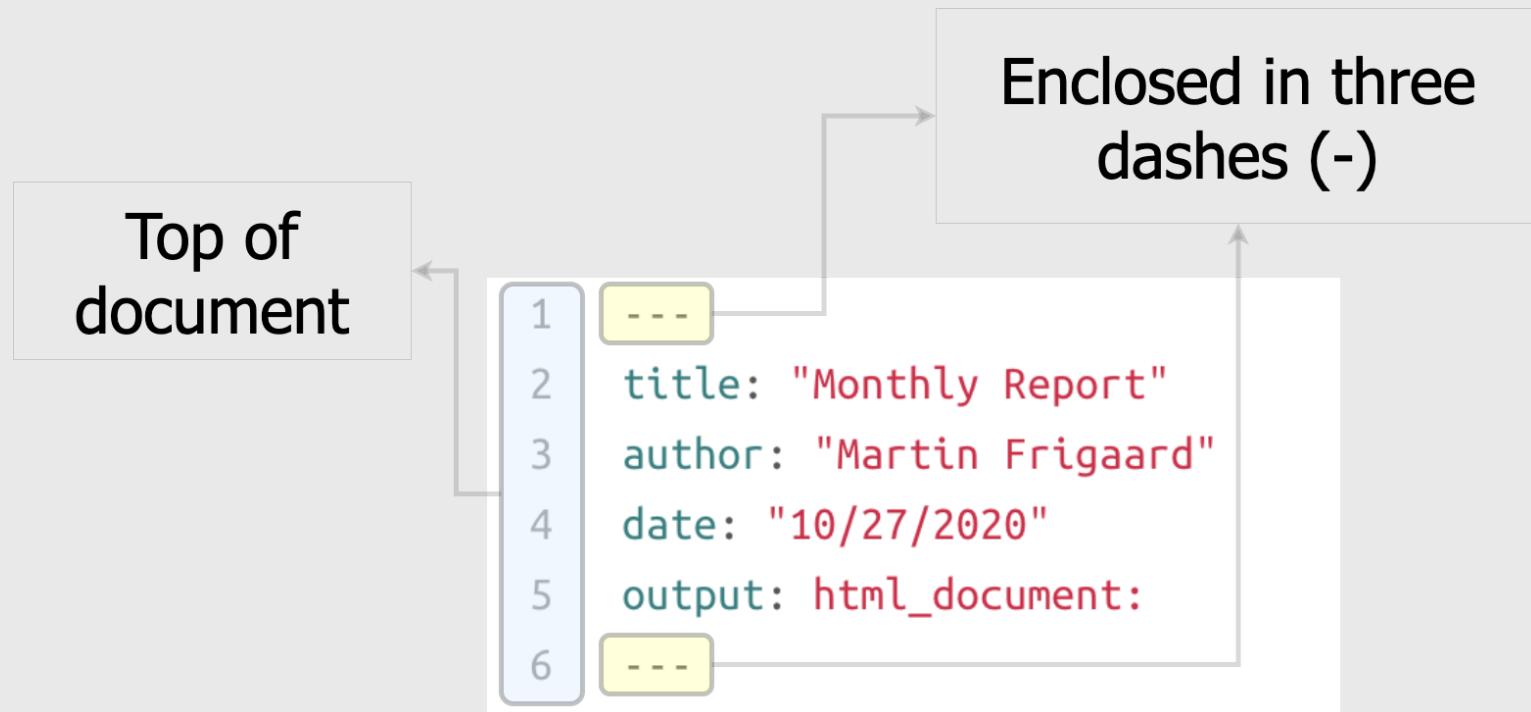


YAML contains the information about the document we're going to create

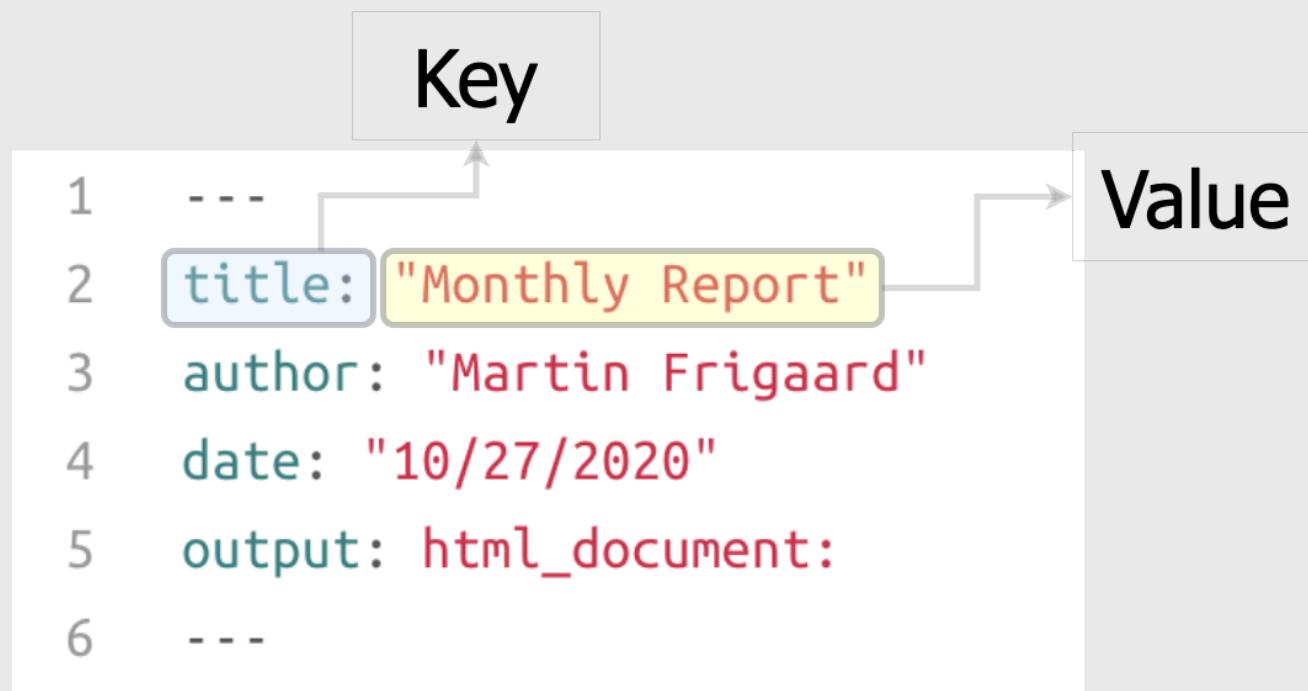
```
---
```

```
title: "Monthly Report"
author: "Martin Frigaard"
date: "10/27/2020"
output: html_document
---
```

# R Markdown: YAML format



# R Markdown: YAML format



# R Markdown: YAML



There are many YAML arguments and options

*Indentation matters in YAML!!*

Check out the [YAML Fieldguide](#) for a comprehensive list



# Example YAML output options

Table of contents:

`toc`: logical (`true` or `false`)

`toc_float`: logical (`true` or `false`)

`toc_depth`: set numerically 0 - 6

# Exercise 2: create a floating table of contents

Change the **output** in the YAML header to the following:

```
output:  
  html_document:  
    toc: yes  
    toc_float: true
```

***Knit the document again***



# YAML output options: table of contents

## Floating table of contents (rendered)

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: true  
9 ---
```

R Markdown  
Including Plots

## Monthly Report

Martin Frigaard

10/27/2020

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:



# Exercise 3: text highlighting and themes

Add the following two options to your YAML header

```
output:  
  html_document:  
    toc: yes  
    toc_float: yes  
    highlight: zenburn  
    theme: united
```

***Knit the document again***



# YAML: text highlighting and theme options

Text highlighting and theme options (rendered)

The diagram illustrates the connection between the YAML front matter of an R Markdown document and its rendered output. On the left, the YAML code is shown:

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: yes  
9     highlight: zenburn  
10    theme: united  
11 ---
```

Two arrows point from the 'highlight: zenburn' and 'theme: united' lines in the YAML to their corresponding settings in the rendered output window. The rendered output shows the following:

**R Markdown**

**Monthly Report**  
Martin Frigaard  
10/27/2020

**R Markdown**

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

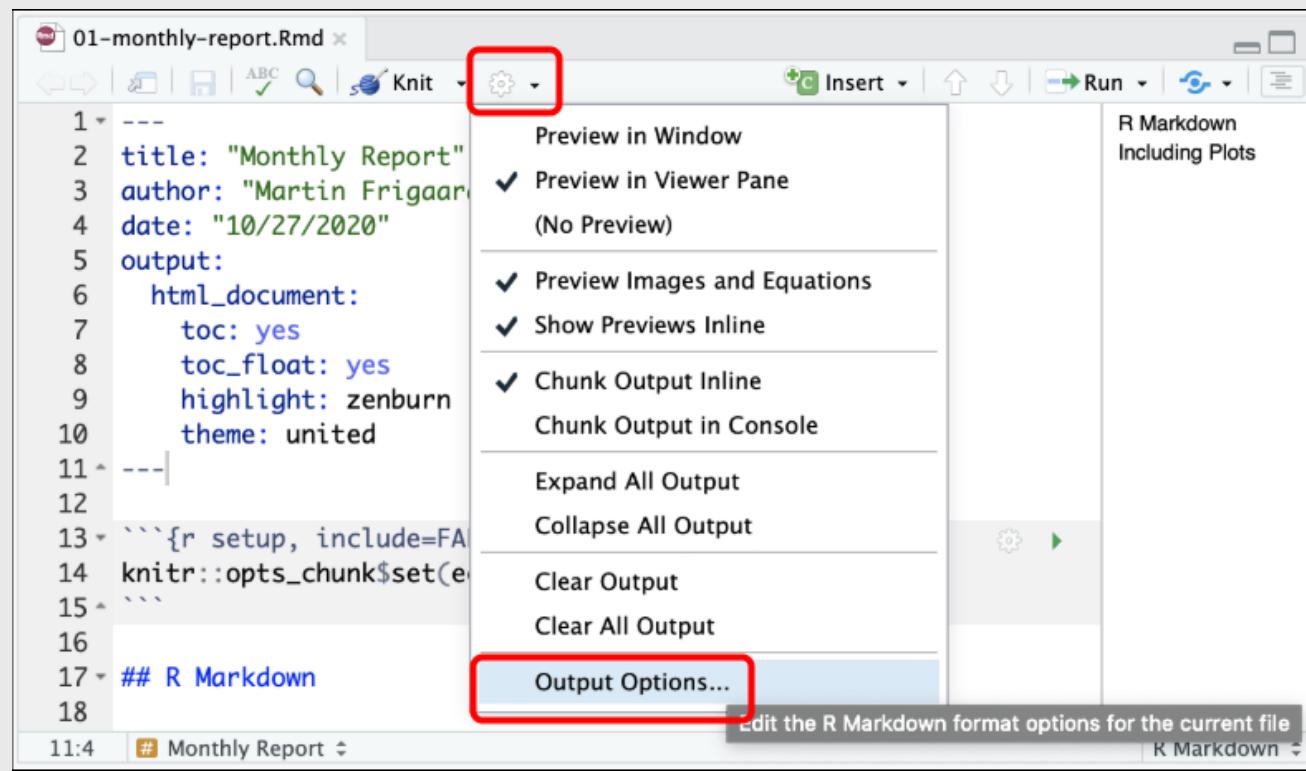
```
summary(cars)
```

	speed	dist
## Min.	4.0	2.00
## 1st Qu.	12.0	1st Qu.: 26.00
## Median	15.0	Median : 36.00
## Mean	15.4	Mean : 42.98
## 3rd Qu.	19.0	3rd Qu.: 56.00
## Max.	25.0	Max. : 120.00



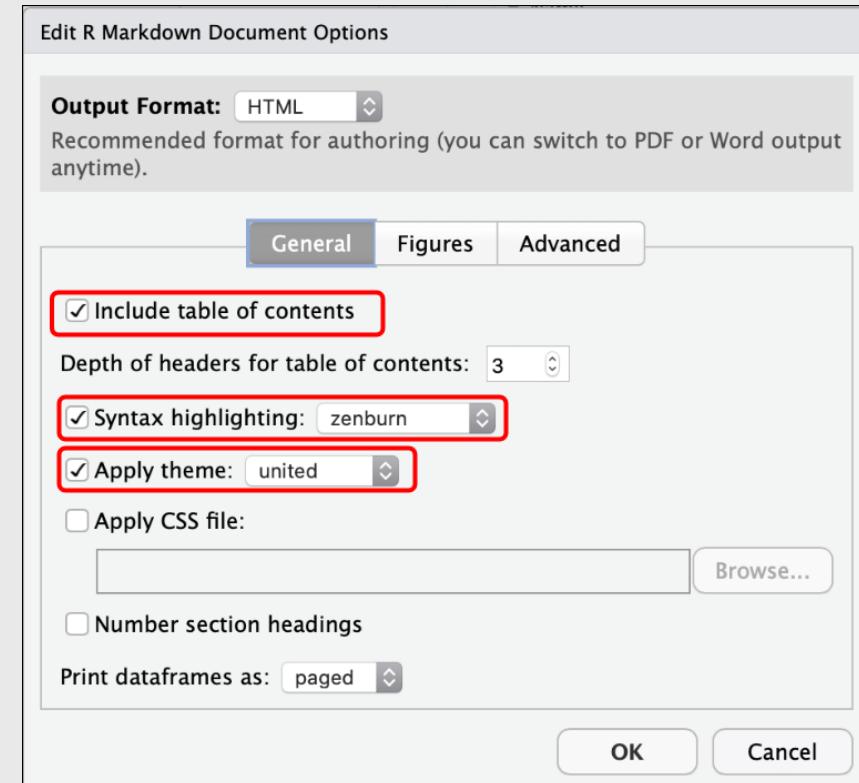
# More YAML options

You can change the YAML contents using the settings (small gear)



# Edit R Markdown Document Options

**This window gives us the ability to manually change some of the YAML settings (but not all of them!)**



# YAML Parameters

YAML parameters can be referred to throughout the document

## ***Create params in YAML header***

```
params:  
  param1: x  
  param2: y  
  data: df
```

## ***Refer to params in .Rmd document***

```
params$param1  
params$param2  
params$data
```



# Exercise 4: Using YAML parameters

***Add the following `params` option in the YAML header***

```
params:  
  small_pressure: !r  
head(pressure)
```

***Add this code to the end of the document***

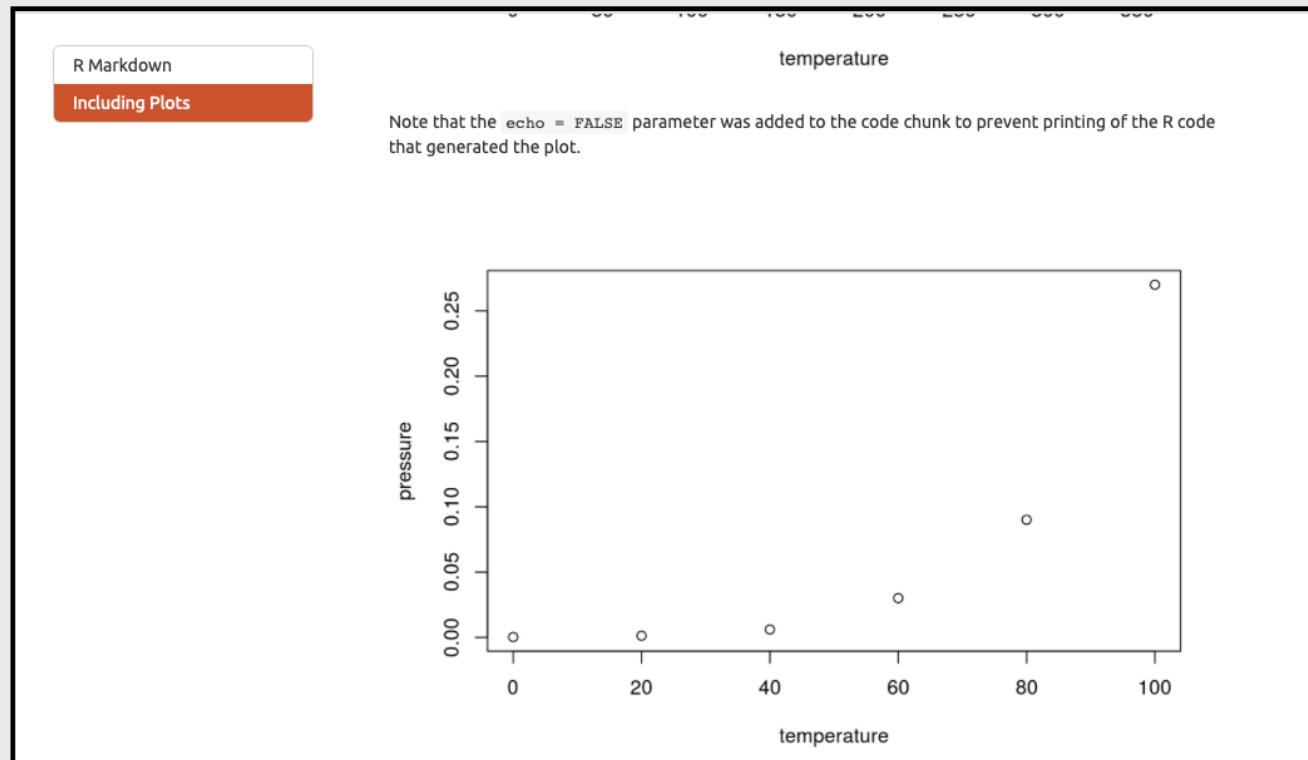
```
```{r small_pressure, echo=FALSE}  
plot(params$small_pressure)  
```
```

***Knit the document again!***



# See our new plot with the `params`

We can see the new plot with the reduced sample size



# YAML output formats



| Function                               | Outputs             |
|--|---------------------|
| <code>html_document()</code>           | HTML document       |
| <code>pdf_document()</code>            | PDF document        |
| <code>word_document()</code>           | Word document       |
| <code>odt_document()</code>            | ODT document        |
| <code>rtf_document()</code>            | RTF document        |
| <code>md_document()</code>             | Markdown document   |
| <code>slidy_presentation()</code>      | Slidy Slides (HTML) |
| <code>beamer_presentation()</code>     | Beamer Slides (PDF) |
| <code>ioslides_presentation()</code>   | ioslides (HTML)     |
| <code>powerpoint_presentation()</code> | PowerPoint (pptx)   |

# R Markdown

~~▪ `.yaml` = Metadata~~

~~▪ `.md` = Prose~~

~~▪ `.R` = Code~~



# Basic Markdown Syntax



## Italics & Bold

```
*italic*    **bold**  
italic    bold
```

*italic* **bold**  
*italic* **bold**

# Basic Markdown Syntax



## Headers

```
# Header 1  
## Header 2  
### Header 3
```

Header 1

Header 2

Header 3

# Basic Markdown Syntax



## Bullets & Numeric Lists

```
* Item 1
* Item 2
  + Item 2a
  + Item 2b
```

```
1. Item 1
2. Item 2
```

- Item 1
  - Item 2
    - Item 2a
    - Item 2b
1. Item 1
  2. Item 2

# Basic Markdown Syntax



## Hyperlinks

*becomes...*

```
https://www.biomarin.com/
```

```
[linked phrase] (https://www.biomarin.com/)
```

linked phrase

# Basic Markdown Syntax



Images

*becomes...*

```

![optional caption] (https://www.r-
project.org/logo/Rlogo.png)
```



# Basic Markdown Syntax



## Math Equations

```
$equation$
```

```
$$ equation $$
```

*equation*

*equation*

# Basic Markdown Syntax



## Super scripts & Strike-through

```
superscript^2^  
~~strikethrough~~
```

superscript^2^

~~strikethrough~~

# Exercise 5: Markdown Formatting

Delete the top portion of the markdown in `01-monthly-report.Rmd`.

Delete {

```
1 - ---
2   title: "Monthly Report"
3   author: "Martin Frigaard"
4   date: "10/27/2020"
5   output:
6     html_document:
7       toc: yes
8       toc_float: yes
9       highlight: zenburn
10      theme: united
11   params:
12     small_pressure: !r head(pressure)
13 ^ ---
14
15 ^ ``{r setup, include=FALSE}
16 knitr::opts_chunk$set(echo = TRUE)
17 ^
18
19 ^ ## R Markdown
20
21 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF,
and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
22
23 When you click the **Knit** button a document will be generated that includes both content as
well as the output of any embedded R code chunks within the document. You can embed an R code
chunk like this:
```



# Exercise 5: Markdown Formatting

Add the text below to your report

*This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.*

Include the following formatting:

1. make `monthly report` italic
2. include this hyperlink for `Rmarkdown`:  
<https://rmarkdown.rstudio.com/>
3. format `code` as code.

***Knit the document when you're finished***



# Exercise 5: Markdown Formatting (rendered)

The screenshot shows the RStudio interface with an R Markdown file named "01-monthly-report.Rmd" open on the left. The code editor contains R Markdown syntax, including YAML front matter, R code chunks, and a summary text block. The right pane displays the rendered output, which includes a title section, an R Markdown section, and a code block with a summary of the "cars" dataset.

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: yes  
9     highlight: zenburn  
10    theme: united  
11 params:  
12   small_pressure: !r head(pressure)  
13 ---  
14  
15 ``{r setup, include=FALSE}  
16 knitr::opts_chunk$set(echo = TRUE)  
17 ``  
18  
19 ## R Markdown  
20  
21 This is a *monthly report* generated with  
22 [RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool  
23 for combining text and `code`.  
24 ``{r cars}  
25 summary(cars)  
26 ``
```

**R Markdown**  
Including Plots

# Monthly Report

Martin Frigaard  
10/27/2020

## R Markdown

This is a *monthly report* generated with **RMarkdown**, a literate programming tool for combining text and `code`.

```
summary(cars)
```

|            | speed | dist           |
|------------|-------|----------------|
| ## Min.    | 4.0   | Min. : 2.00    |
| ## 1st Qu. | 12.0  | 1st Qu.: 26.00 |
| ## Median  | 15.0  | Median : 36.00 |
| ## Mean    | 15.4  | Mean : 42.98   |
| ## 3rd Qu. | 19.0  | 3rd Qu.: 56.00 |
| ## Max.    | 25.0  | Max. : 120.00  |



# Exercise 6: Tabssets



Remove the `toc` and `toc_float` options from your YAML header

```
output:  
  html_document:  
    highlight: zenburn  
    theme: united  
params:  
  small_pressure: !r head(pressure)
```

# Exercise 6: Tabssets



Make the following changes to the R Markdown header sections

```
18  
19 - ## R Markdown {.tabset}  
20  
21 This is a **monthly report** generated with  
[RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool  
for combining text and `code`.  
22  
23 - ### Summary  
24  
25 - ``{r cars}  
26 summary(cars)  
27 - ``  
28  
29 - ### Including Plots  
30  
31 You can also embed plots, for example:  
32  
33 - ``{r pressure, echo=FALSE}  
34 plot(pressure)  
35 - ``  
36  
37 Note that the `echo = FALSE` parameter was added to the code chunk to  
prevent printing of the R code that generated the plot.  
38  
39 - ``{r small_pressure, echo=FALSE}  
40 plot(params$small_pressure)  
41 - ``
```

Include tabset in curly brackets

Add 'Summary' level 3 header

Change 'Including Plots' to level 3 header

# Exercise 6: Tabsets (rendered)



## Tab 1

The screenshot shows the RStudio interface with an R Markdown file open on the left and its rendered output on the right.

**Code Editor (Left):**

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "11/5/2020"  
5 output:  
6   html_document:  
7     highlight: zenburn  
8     theme: united  
9 params:  
10   small_pressure: !r head(pressure)  
11 ---  
12  
13 ```{r setup, include=FALSE}  
14 knitr::opts_chunk$set(echo = TRUE)  
15 ```  
16  
17 ## R Markdown {.tabset}  
18  
19 This is a monthly report generated with RMarkdown (https://rmarkdown.rstudio.com/), a literate programming tool for combining text and code.  
20  
21 ### Summary  
22  
23 ```{r cars}  
24 summary(cars)  
25 ```  
26  
27 ### Including Plots  
28  
29 You can also embed plots, for example:  
30  
31 ```{r pressure, echo=FALSE}  
32 plot(pressure)  
33  
34 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R  
35 code that generated the plot.  
36  
37 ```{r small_pressure, echo=FALSE}  
38 plot(params$small_pressure)  
39  
40  
41
```

**R Markdown Preview (Right):**

## Monthly Report

Martin Frigaard  
11/5/2020

### R Markdown

This is a **monthly report** generated with [RMarkdown](#), a literate programming tool for combining text and code.

[Summary](#) [Including Plots](#)

```
summary(cars)
```

```
##      speed      dist  
##  Min.   :4.0   Min.   : 2.00  
##  1st Qu.:12.0  1st Qu.:26.00  
##  Median :15.0  Median :36.00  
##  Mean   :15.4  Mean   :42.98  
##  3rd Qu.:19.0  3rd Qu.:56.00  
##  Max.   :25.0  Max.   :120.00
```

# Exercise 6: Tabsets (rendered)



## Tab 2

The screenshot shows the RStudio interface with an R Markdown file open on the left and its rendered output on the right.

**Code Editor (Left):**

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "11/5/2020"  
5 output:  
6   html_document:  
7     highlight: zenburn  
8     theme: united  
9 params:  
10   small_pressure: !r head(pressure)  
11 ---  
12  
13 ```{r setup, include=FALSE}  
14 knitr::opts_chunk$set(echo = TRUE)  
15 ```  
16  
17 ## R Markdown {.tabset}  
18  
19 This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.  
20  
21 ## Summary  
22  
23 ```{r cars}  
24 summary(cars)  
25 ```  
26  
27 ## Including Plots  
28  
29 You can also embed plots, for example:  
30  
31 ```{r pressure, echo=FALSE}  
32 plot(pressure)  
33 ```  
34  
35 Note that the 'echo = FALSE' parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
36  
37 ```{r small_pressure, echo=FALSE}  
38 plot(params$small_pressure)  
39 ```
```

**R Markdown Preview (Right):**

## Monthly Report

Martin Frigaard  
11/5/2020

### R Markdown

This is a **monthly report** generated with [RMarkdown](#), a literate programming tool for combining text and `code`.

[Summary](#) [Including Plots](#)

You can also embed plots, for example:

A scatter plot showing the relationship between temperature (x-axis, ranging from 0 to 350) and pressure (y-axis, ranging from 0 to 800). The data points show a strong positive correlation, with pressure increasing as temperature increases. There are approximately 20 data points plotted.

| temperature | pressure |
|-------------|----------|
| 10          | 10       |
| 20          | 10       |
| 30          | 10       |
| 40          | 10       |
| 50          | 10       |
| 60          | 10       |
| 70          | 10       |
| 80          | 10       |
| 90          | 10       |
| 100         | 10       |
| 110         | 10       |
| 120         | 10       |
| 130         | 10       |
| 140         | 10       |
| 150         | 10       |
| 160         | 10       |
| 170         | 10       |
| 180         | 10       |
| 190         | 10       |
| 200         | 10       |
| 210         | 10       |
| 220         | 10       |
| 230         | 10       |
| 240         | 10       |
| 250         | 10       |
| 260         | 10       |
| 270         | 10       |
| 280         | 10       |
| 290         | 10       |
| 300         | 10       |
| 310         | 10       |
| 320         | 10       |
| 330         | 10       |
| 340         | 10       |
| 350         | 800      |

# R Markdown

~~.yaml = Metadata~~

~~.md = Prose~~

.R = Code



# Code chunks (`setup`)



The first bit of R code in our RMarkdown file is the `setup` chunk

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
````
```

Chunks named '`setup`' are special because they can set global options

'`include=FALSE`' means this code is run, but not displayed

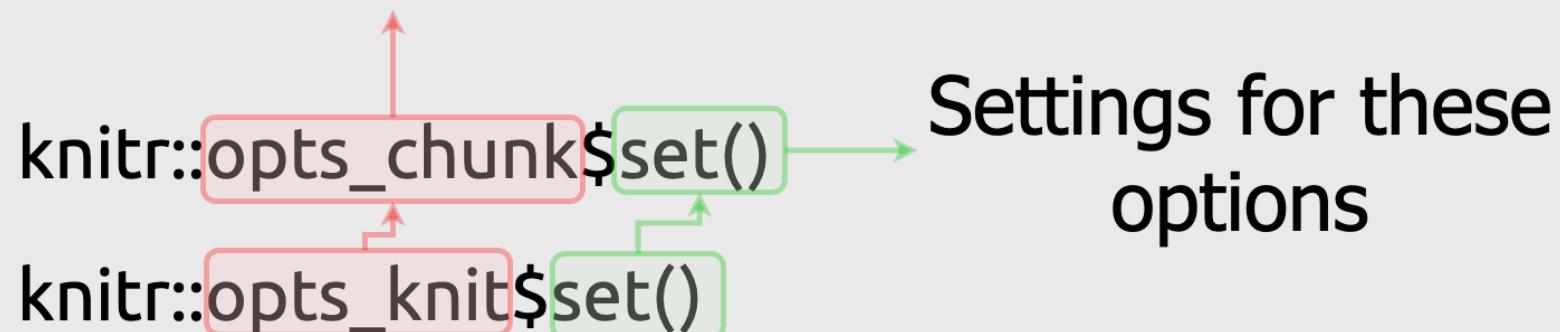


# Code chunks (setup)

R Markdown document options come from the `knitr` package

We can access both the with syntax below:

Code chunks and  
knit options



# Code chunks (`setup`)



The `echo=TRUE` option controls whether we want to display the code in the code chunk

Other common options regarding code are `eval`, `tidy`, `error`, `message`, and `warning`

Advanced options can control language engines (`engine`), caching (`cache`, `dependson`), and plot animations (`fig.show`)

# Code chunks (`setup`)



Many options for code chunks

| Option               | Document Effect                               |
|----------------------|---|
| <code>include</code> | run code, but do/don't print code and results |
| <code>eval</code>    | do/don't evaluate the code                    |
| <code>echo</code>    | run code, but don't print code                |
| <code>message</code> | do/don't print messages (e.g. from functions) |
| <code>warning</code> | do/don't print warnings                       |

# Code chunks



```
1   ```{r pressure, echo=FALSE}
2   plot(pressure)
3   ````
```

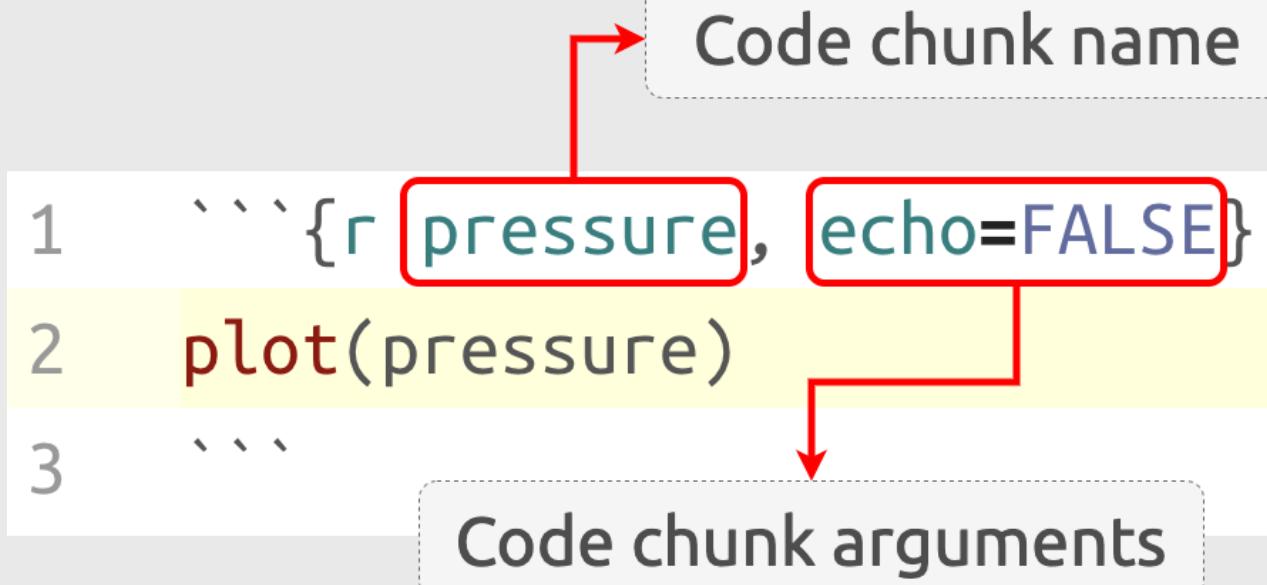
# Code chunk fences



Code chunk fences

```
1 ````{r pressure, echo=FALSE}
2 plot(pressure)
3 ````
```

# Code chunk names and arguments

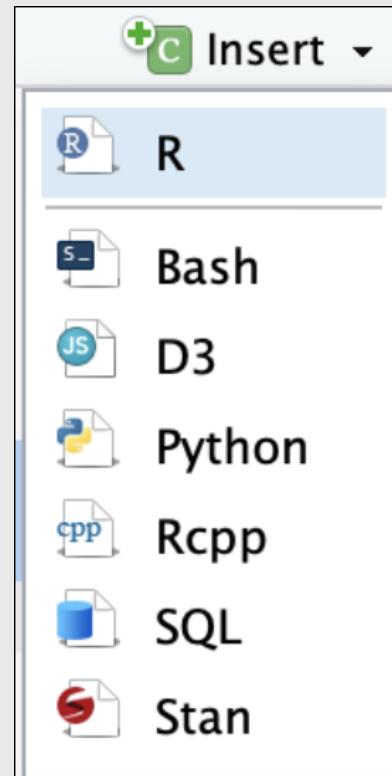


See the [knitr web page](#) for complete list of options

# Inserting code chunks



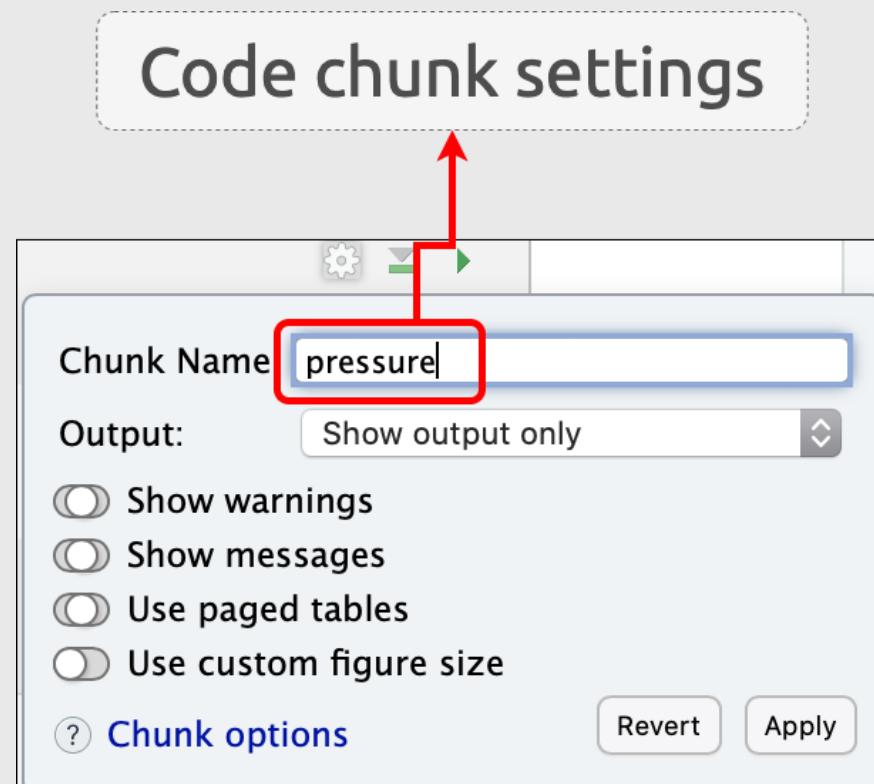
Use keyboard shortcuts **CMD/CTRL + I** or **ALT/OPTION + I**





# Edit code chunk options

You can edit code chunk options using the icon (small gear)



# Code Chunk Engines



## More and more code engines all the time

```
names(knitr:::knit_engines$get())
```

```
[1] "awk"        "bash"       "coffee"      "gawk"       "groovy"     "haskell"
[7] "lein"       "mysql"      "node"        "octave"     "perl"       "pgsql"
[13] "Rscript"    "ruby"       "sas"         "scala"      "sed"        "sh"
[19] "stata"      "zsh"        "highlight"   "Rcpp"       "tikz"       "dot"
[25] "c"          "cc"         "fortran"     "fortran95" "asy"        "cat"
[31] "asis"       "stan"       "block"       "block2"     "js"         "css"
[37] "sql"        "go"         "python"      "julia"      "sass"       "scss"
[43] "R"          "bslib"      "targets"
```

# Exercise 7: code chunks (kable)

Create a new **Tables** level three header under the **Summary** heading,

```
### Tables
```

Insert the following code chunk under **Tables**

```
```{r kable}
knitr::kable(params$small_pressure)
```
```

*insert the code block manually with the keyboard short-cut, or use the "Insert" button*



# Exercise 7: code chunks (kable rendered)



We can see the `small_pressure` parameter from the YAML has been rendered in the new **Tables** tab. `kable` tables are great for presenting small, summary tables.

The diagram illustrates the rendering process of an R Markdown code chunk. On the left, a code editor window shows the following R Markdown code:

```
### Tables
```{r kable}
knitr::kable(params$small_pressure)
```

```

A red box highlights the `knitr::kable(params$small_pressure)` line. A red arrow points from this line to the rendered output on the right. The right side shows the generated R Markdown page with the following content:

R Markdown  
This is a **monthly report** generated with **RMarkdown**, a literate programming tool for combining text and `code`.  
Summary   Tables   Including Plots

```
knitr::kable(params$small_pressure)
```

|     | temperature | pressure |
|-----|-------------|----------|
| 0   | 0.0002      |          |
| 20  | 0.0012      |          |
| 40  | 0.0060      |          |
| 60  | 0.0300      |          |
| 80  | 0.0900      |          |
| 100 | 0.2700      |          |

Read more about `kable` table options [here](#)



# Exercise 8: code chunks (paged)

We are going to repeat the process above, but with a larger table (`mtcars`)

Insert the following code chunk above the `knitr::kable()` output:

```
### Tables

```{r paged}
rmarkdown::paged_table(mtcars)
```

```{r kable}
knitr::kable(params$small_pressure)
```
```

Knit the document

# Exercise 8: code chunk (paged rendered)

Paged tables are great for larger datasets

The diagram illustrates the process of rendering paged tables from R Markdown code. On the left, a screenshot of an R Markdown editor shows two code chunks. The top chunk contains `r paged` code to render a paged table of the mtcars dataset. The bottom chunk uses `r kable` to render a standard table of small pressure parameters. A red box highlights the `r paged` code in the top chunk, and a red arrow points from this box to the rendered paged table on the right. The right side shows the rendered output in an R Markdown viewer. The header says "R Markdown" and "This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code." Below the header are tabs for "Summary", "Tables" (which is selected), and "Including Plots". The main content area displays the rendered paged table of mtcars data, showing columns for mpg, cyl, disp, hp, drat, wt, qsec, vs, and am. The table is paginated, with rows 1 through 10 shown. At the bottom of the table, there is a footer with navigation links for "Previous" (page 1), "2", "3", "4", and "Next".

| mpg   | cyl   | disp  | hp    | drat  | wt    | qsec  | vs    | am    |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| <dbl> |
| 21.0  | 6     | 160.0 | 110   | 3.90  | 2.620 | 16.46 | 0     | 1     |
| 21.0  | 6     | 160.0 | 110   | 3.90  | 2.875 | 17.02 | 0     | 1     |
| 22.8  | 4     | 108.0 | 93    | 3.85  | 2.320 | 18.61 | 1     | 1     |
| 21.4  | 6     | 258.0 | 110   | 3.08  | 3.215 | 19.44 | 1     | 0     |
| 18.7  | 8     | 360.0 | 175   | 3.15  | 3.440 | 17.02 | 0     | 0     |
| 18.1  | 6     | 225.0 | 105   | 2.76  | 3.460 | 20.22 | 1     | 0     |
| 14.3  | 8     | 360.0 | 245   | 3.21  | 3.570 | 15.84 | 0     | 0     |
| 24.4  | 4     | 146.7 | 62    | 3.69  | 3.190 | 20.00 | 1     | 0     |
| 22.8  | 4     | 140.8 | 95    | 3.92  | 3.150 | 22.90 | 1     | 0     |
| 19.2  | 6     | 167.6 | 123   | 3.92  | 3.440 | 18.30 | 1     | 0     |



# Exercise 8: paged tables



## R Markdown

This is a **monthly report** generated with RMarkdown, a literate programming tool for combining text and `code`.

Summary

Tables

Including Plots

```
rmarkdown::paged_table(mtcars)
```

|                   | mpg   | cyl   | disp  | hp    | drat  | wt    | qsec  | vs    | am    | ▶ |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|---|
|                   | <dbl> |   |
| Mazda RX4         | 21.0  | 6     | 160.0 | 110   | 3.90  | 2.620 | 16.46 | 0     | 1     |   |
| Mazda RX4 Wag     | 21.0  | 6     | 160.0 | 110   | 3.90  | 2.875 | 17.02 | 0     | 1     |   |
| Datsun 710        | 22.8  | 4     | 108.0 | 93    | 3.85  | 2.320 | 18.61 | 1     | 1     |   |
| Hornet 4 Drive    | 21.4  | 6     | 258.0 | 110   | 3.08  | 3.215 | 19.44 | 1     | 0     |   |
| Hornet Sportabout | 18.7  | 8     | 360.0 | 175   | 3.15  | 3.440 | 17.02 | 0     | 0     |   |
| Valiant           | 18.1  | 6     | 225.0 | 105   | 2.76  | 3.460 | 20.22 | 1     | 0     |   |
| Duster 360        | 14.3  | 8     | 360.0 | 245   | 3.21  | 3.570 | 15.84 | 0     | 0     |   |
| Merc 240D         | 24.4  | 4     | 146.7 | 62    | 3.69  | 3.190 | 20.00 | 1     | 0     |   |
| Merc 230          | 22.8  | 4     | 140.8 | 95    | 3.92  | 3.150 | 22.90 | 1     | 0     |   |
| Merc 280          | 19.2  | 6     | 167.6 | 123   | 3.92  | 3.440 | 18.30 | 1     | 0     |   |

1-10 of 32 rows | 1-10 of 12 columns

Previous **1** **2** **3** **4** Next

# Inline R Code



R Markdown also supports inline R code

Grave accent or back-tick  
around code

Some text `r` `function(x)` more text

A diagram illustrating inline R code. The text "Some text" is in gray, and "more text" is in gray. In the center, the code "r function(x)" is highlighted with a blue dotted box. Two red arrows point upwards from the bottom of the slide to the grave accents (the ' character) surrounding the word "function".

# Inline R Code



Inline R code allows us to include summaries of our analysis in the report

Some text `r **function(x)**` more text



Any R expression

# Exercise 9: Add Inline Code



We're going to add a Pearson correlation between speed and stopping distance

**Include the following code under the *Summary* level three header**

The correlation between speed and stopping distance is 0.8068949

**Knit the document again**

# Exercise 9: Add Inline Code (rendered)



```
### Summary  
  
The correlation between speed and stopping distance  
is `r cor(x = cars$speed, y = cars$dist)`  
  
```{r cars}  
summary(cars)  
```
```

## Monthly Report

Martin Frigaard

10/27/2020

### R Markdown

This is a **monthly report** generated with **RMarkdown**, a literate programming tool for combining text and **code**.

Summary   Tables   Including Plots

The correlation between speed and stopping distance is 0.8068949

```
summary(cars)
```

```
##      speed          dist
##  Min.   : 4.0   Min.   : 2.00
##  1st Qu.:12.0  1st Qu.: 26.00
##  Median :15.0  Median : 36.00
##  Mean   :15.4  Mean   : 42.98
##  3rd Qu.:19.0  3rd Qu.: 56.00
##  Max.   :25.0  Max.   :120.00
```

# Make cool stuff in R Markdown!

bookdown

blogdown

these slides!



# Resources



- **YAML:** check out the [yamlthis package](#) for tools and documentation for working with YAML
- **Markdown:** [Commonmark](#) has a quick ten-twenty minute tutorial on markdown.
- [R Markdown](#): A comprehensive but friendly introduction to R Markdown and friends. Free online!
- [R for Data Science](#): A comprehensive but friendly introduction to the tidyverse. Free online.
- [R Markdown for Scientists](#): R Markdown for Scientists workshop material